CLAIM AMENDMENTS

Claim Amendment Summary

Claims pending

- Before this Amendment: Claims 1, 3-7, 10, 11, 14-20, 22, 24-27, 29-31 and 39-54.
- After this Amendment: Claims 26-27, 29, 31, 39-40, 51, and 53-54.

Non-Elected, Canceled, or Withdrawn claims: Claims 1-25, 28, 30, 32-38, 41-50 and 52.

Amended claims: Claims 26, 29, 31, 51 and 53.

New claims: None.

Claims:

1 - 25. (Canceled)

26. (Currently Amended) A computer apparatus comprising a computing device and memory:

a code unit;

a composable set of rewriters, each rewriter configured to rewrite the code unit in a unique manner;

a rewrite manager configured to identify one or more rewriters from the composable set of rewriters, sequence the rewriters and to execute the identified one or more rewriters against the code unit, wherein the rewrite manager ~~determines the identity and sequence by~~ accessing a rewrite list that identifies which rewriter to apply to rewrite the code unit and provides the sequence the rewriters are applied to the code unit; and

a rewritten code unit generated by executing the identified one or more rewriters against the code unit;

a first digital signature associated with the code unit; and

a set of second digital signatures, each second digital signature associated with a specific rewriter from the composable set of rewriters;

wherein the rewrite manager is further configured to determine if the code unit is trusted based on the first digital signature, determine if each rewriter from the identified one or more rewriters is trusted based on a corresponding second digital signature from the set of second digital signatures, and execute the identified one or more rewriters against the code unit only if both the code unit and each rewriter from the identified one or more rewriters are trusted.

27. (Previously Presented) A computer apparatus as recited in claim 26, further comprising a rewrite cache, the rewrite manager further configured to store the rewritten code unit in the rewrite cache.

28. (Canceled)

29. (Currently Amended) A computer apparatus as recited in claim [[28]] 26, wherein the rewriter list is a component selected from a group comprising:

a list of rewriters in the code unit;

a list of rewriters in a stand-alone file;

a list of rewriters in a security policy; and

a list of rewriters in an installation tool.

30. (Canceled)

31. (Currently Amended) A computer apparatus as recited in claim [[30]] 26, further comprising a third digital signature associated by the rewrite manager with the rewritten code unit and configured to verify that the rewritten code unit is trusted.

39. (Previously Presented) A computer apparatus as recited in claim 26, wherein the rewrite manager is a component selected from a group comprising:

a stand-alone rewrite module;

a rewrite module configured as part of an operating system;

a rewrite module configured as part of an installation tool; and

a rewrite module configured as part of a security policy.

40. (Previously Presented) A computer apparatus as recited in claim 26, selected from a group comprising:

a development computer configured to develop the code unit;

an intermediate computer configured to install the code unit; and

a deployment computer configured to execute the code unit.

41 - 50. (Canceled).

51. (Currently Amended) A method comprising:

receiving a code unit;

determining that the code unit is to be rewritten by a rewriter, wherein the rewriter is determined by a rewrite manager that accesses a rewrite list that identifies which rewriter to apply to rewrite the code unit and provides the sequence the rewriters are applied to the code unit;

determining if the code unit and the rewriter are trusted;

running the rewriter against the code unit to generate a rewritten code unit if the code unit and the rewriter are trusted;

generating a digital signature for the rewritten code unit;

attaching the digital signature to the rewritten code unit; and

storing the rewritten code unit in a cache.

52. (Canceled)

53. (Currently Amended) A method as recited in claim [[52]] 51, further comprising:

receiving a call to execute the code unit;

recognizing that the code unit has been rewritten;

loading the rewritten code unit from the cache;

verifying the digital signature attached to the rewritten code unit; and

executing the rewritten code unit if the verifying indicates that the rewritten code unit is secure.

54. (Original) A method as recited in claim 51, wherein the rewriter is an application compatibility rewriter and the determining that the code unit is to be rewritten comprises:

identifying the code unit as an application; and

consulting an application compatibility rewrite database to determine if any part of the application needs to be rewritten for compatibility with a current execution environment.